

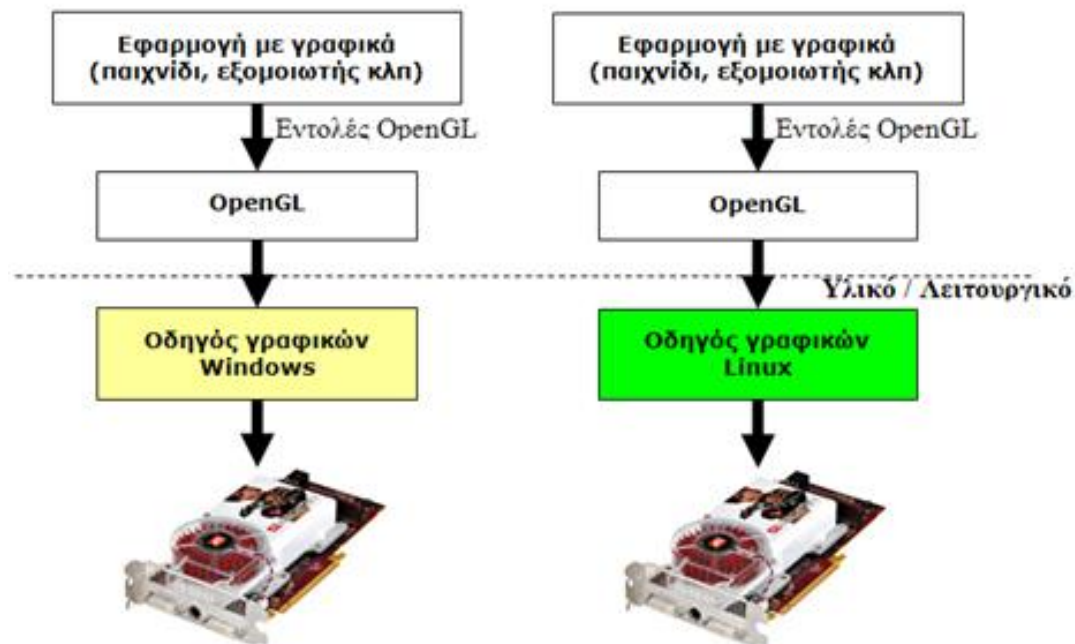
# Εισαγωγή στην OpenGL: μέρος 1ο

Θα κάνουμε μια μικρή εισαγωγή στην OpenGL, η οποία μπορεί να χρησιμοποιηθεί για να αναπτυχθούν παιχνίδια.

## Τι είναι η OpenGL

Η OpenGL είναι ένα σύνολο εντολών (**A**pplication **P**rogramming **I**nterface – API) που μας επιτρέπει την δημιουργία τρισδιάστατων γραφικών. Δεν είναι γλώσσα προγραμματισμού αλλά μπορεί να χρησιμοποιηθεί με μια πληθώρα γλωσσών προγραμματισμού (C, C++, Java και άλλες) και σε μια πληθώρα λειτουργικών συστημάτων (Windows, Unix, Linux, Mac OS).

Προσφέρει πάνω από 300 εντολές για δημιουργία γραφικών και δρα σαν ένα ενδιάμεσο στρώμα ανάμεσα στην εφαρμογή και στη κάρτα γραφικών που θα αναλάβει τα απεικονίσει τα γραφικά στην οθόνη, κρύβοντας λεπτομέρειες υλοποίησης του υλικού και των οδηγών του.

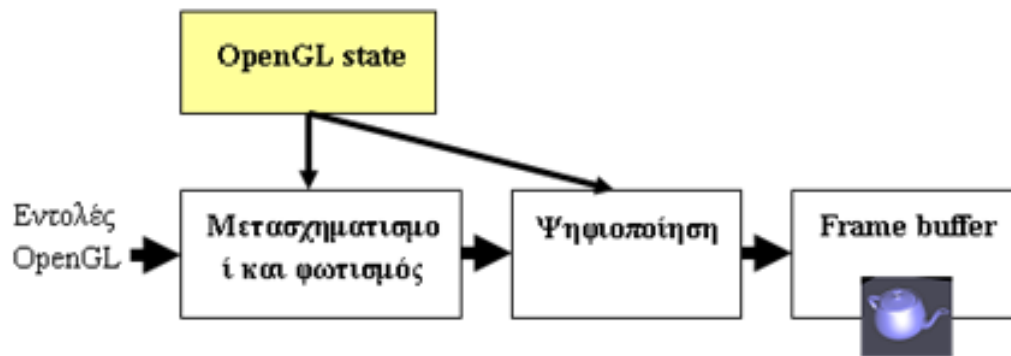


Προγράμματα που χρησιμοποιούν OpenGL μπορούν να τρέξουν και χωρίς επιταχυντή γραφικών (3D graphics card), θα είναι όμως πολύ πιο αργά και πιθανώς να μην υποστηρίζουν όλες τις λειτουργίες της OpenGL.

Η OpenGL είναι το πρώτο ευρέως διαδεδομένο API για γραφικά και υπάρχει σαν στάνταρντ από το 1992. Άλλο ευρέως διαδεδομένο API γραφικών είναι το DirectX της Microsoft το οποίο είναι γραμμένο αποκλειστικά για πλατφόρμες Windows και Xbox/Xbox360.

## OpenGL graphics pipeline

Ας δούμε σχηματικά τα βήματα που κάνει η OpenGL από την στιγμή που πάρει τις εντολές γραφικών από την εφαρμογή μας μέχρι την στιγμή δημιουργίας της τελικής εικόνας (rendered image).



Η OpenGL χρησιμοποιεί μια **μηχανή καταστάσεων** (state machine) για να επικοινωνεί με την εφαρμογή. Σε αυτή την μηχανή καταστάσεων η OpenGL παραμένει διαρκώς σε μια κατάσταση μέχρι να αλλάξει η εφαρμογή την κατάσταση. Παράδειγμα αν θέσουμε το χρώμα που θα χρησιμοποιεί η OpenGL για να ζωγραφίσει ένα μοντέλο, το χρώμα θα παραμείνει στη μνήμη και θα χρησιμοποιείται μέχρι να το αλλάξουμε ή να κλείσουμε την εφαρμογή.

Εφόσον λοιπόν η εφαρμογή καθορίσει το περιβάλλον που θα χρησιμοποιήσει η OpenGL για να απεικονίσει το μοντέλο μας (χρώματα, υφές, πηγές φωτός, κάμερα κλπ), περνάει στο πρώτο στάδιο κατά το οποίο θα μετασχηματίσει και θα φωτίσει (transform and lighting) τα σημεία(vertices) του μοντέλου.

Στην συνέχεια η OpenGL περνά στο στάδιο της ψηφιοποίησης το οποίο λαμβάνει όλες τις πληροφορίες και την γεωμετρία από το προηγούμενο στάδιο (του μετασχηματισμού) και παράγει την τελική, ψηφιακή, εικόνα. Η εικόνα αντιγράφεται στην μνήμη του frame buffer και φτάνει έτσι στην οθόνη του υπολογιστή.

**Αυτή η ακολουθία βημάτων που ακολουθεί η OpenGL για να απεικονίσει το μοντέλο λέγεται graphics pipeline (διασωλήνωση)**

## Τι είναι το GLUT

Η OpenGL όπως είπαμε είναι ένας γρήγορος και ευέλικτος τρόπος να επικοινωνούμε με το hardware γραφικών του υπολογιστή χωρίς να ενδιαφερόμαστε για τις λεπτομέρειες υλοποίησης του. Όμως δεν προσφέρει καθόλου λειτουργίες GUI (Graphical User Interface), δηλαδή δεν έχει την δυνατότητα να ανοίξει και να κλείσει παράθυρα στο λειτουργικό σύστημα, να ζωγραφίσει σε αυτά, ούτε να καταλάβει το πάτημα ενός πλήκτρου ή την κίνηση του ποντικιού, ούτε μπορεί να διαβάσει ένα αρχείο από το δίσκο.

Αυτό έγινε επί σκοπού, μιας και η OpenGL σχεδιάστηκε να τρέχει σε πολλά λειτουργικά συστήματα τα οποία έχουν το δικό τους τρόπο επικοινωνίας με την οθόνη, το ποντίκι, το δίσκο, το πληκτρολόγιο.

Για να γίνει αυτό θα πρέπει να χρησιμοποιήσουμε απευθείας τις εντολές του λειτουργικού μας συστήματος (Win32 εντολές στην περίπτωση των Windows). Εναλλακτικά μπορούμε να χρησιμοποιήσουμε μια από τις έτοιμες βιβλιοθήκες εντολών που υπάρχουν και που θα κάνουν αυτές τις λειτουργίες για εμάς εύκολα.

Μια από τις πιο διαδεδομένες βιβλιοθήκες για αυτό το σκοπό είναι το GLUT (OpenGL Utility Toolkit) το οποίο είναι και αυτό σχεδιασμένο να τρέχει σε πολλά λειτουργικά συστήματα.

Το GLUT προσφέρει ένα σύνολο εντολών που αναλαμβάνουν να ανοίξουν και να κλείσουν εύκολα παράθυρα, να καταγράψουν το πάτημα ενός πλήκτρου ή την κίνηση του ποντικιού. Δεν είναι κατάλληλο να γράψουμε μια κανονική εφαρμογή με αυτό (πχ ένα παιχνίδι) όμως είναι πολύ κατάλληλο για εκπαιδευτικές και πρότυπες εφαρμογές.

## Μορφή εντολών OpenGL/GLUT

Η OpenGL χρησιμοποιεί μια απλή, στάνταρντ, μορφή ονοματολογίας για τις εντολές της. Όλες οι εντολές της έχουν μορφή παρόμοια με:



Οι εντολές του GLUT δεν έχουν κάποιο ιδιαίτερο χαρακτηριστικό εκτός του ότι κάθε εντολή ξεκινά με το πρόθεμα `glut`:

```
glutInitWindowSize(Width, Height);
```

## Τύποι δεδομένων OpenGL

Χάριν ομοιομορφίας, και για να μπορέσει να υποστηρίξει διαφορετικές πλατφόρμες, η OpenGL ορίζει μια σειρά από τύπους δεδομένων οι κυριότεροι των οποίων φαίνονται στο παρακάτω πίνακα:

| Τύπος δεδομένων OpenGL | Ορίζεται ως                 | Αντιστοιχία με τύπο δεδομένων στη C | Επίθεμα |
|------------------------|-----------------------------|-------------------------------------|---------|
| GLbyte                 | Ακέραιος 8-bit              | signed char                         | b       |
| GLshort                | Ακέραιος 16-bit             | short                               | s       |
| GLint                  | Ακέραιος 32-bit             | int                                 | i       |
| GLdouble               | Κινητής υποδιαστολής 64-bit | double                              | d       |
| GLubyte                | Θετικός ακέραιος 8-bit      | unsigned char                       | ub      |
| GLushort               | Θετικός ακέραιος 16-bit     | unsigned short                      | us      |
| GLuint                 | Θετικός ακέραιος 32-bit     | unsigned int                        | ui      |
| GLchar                 | Χαρακτήρας 8-bit            | char                                | -       |
| GLfloat                | Κινητής υποδιαστολής 32-bit | float                               | f       |
| GLboolean              | Θετικός ακέραιος 8-bit      | unsigned char                       |         |

Η OpenGL εγγυάται ότι για παράδειγμα ένας GLShort αριθμός θα έχει εύρος 16-bit ανεξάρτητα αν το λειτουργικό σύστημα είναι Windows, Unix ή κάποιο άλλο.

Το επίθεμα είναι το γράμμα που βρίσκουμε στο τέλος μιας OpenGL εντολής και που δείχνει τον τύπο δεδομένων των παραμέτρων.